REMARKS

The Office action dated August 29, 2002 has been carefully considered. In the Office Action, claims 11, 12, 21 and 22 were objected to for informalities, and claims 1-7, and 10-14 were rejected under 35 U.S.C. § 102(a) as being anticipated by Jay Lang, via a reference entitled "IBM Bolsters Windows NT Reliability With Tools Suite -- Package Provides interoperability with other platforms", *Information Week*, July 20, 1998, pp. A6 (hereinafter referred to as "Lang"). Further, claims 8, 9 and 15-24 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lang, in view of Stupek, U.S. Patent No. 5,586,304 (hereinafter Stupek). By the present amendment, claim 23 has been canceled, claims 1-22 and 24 have been amended, claims 25-43 have been added, and the rejections traversed in view of the following remarks. Applicants respectfully submit that the amendments to the claims were made to more particularly point out and distinctly claim the subject matter, not to distinguish the claimed subject matter from the prior art, and maintain that the claims recited patentable subject matter as filed and prior to amendment in view of the following remarks. Reconsideration is respectfully requested.

Regarding the objections, applicants have removed the grammatically incorrect language via the claim amendments. Applicants submit that the objections have been overcome by these amendments.

Assigned, Advertised and Published Programs

Turning to the rejection on the art, the present invention as claimed is generally directed towards at least three significant concepts that are clearly not taught or suggested by the prior art of record, namely 1) assigned programs, 2) assigned programs that are advertised as available for execution prior to their installation, and 3) published programs. Further, the recited manner in which assigned and published programs are deployed, e.g., via an association with a policy recipient, and/or via an advertise script, are also not taught or suggested by the prior art

As described in the specification, an assigned program is one that an administrator or the like, via policy data, declares it mandatory for a policy recipient to have, whether that policy recipient is a computer or a machine. Note that by mandatory, it is meant that the policy recipient has no option to override the assignment (however it is possible for one policy such as that of a different administrator to supersede another policy, and thus configure the computer system in some other manner). At each logon event (for a user) or network connection event (for a machine), or at some other time such as periodically, the computer system is automatically configured according to the policy data to have the assigned programs made available for execution on the computer system. Because the computer will be reconfigured as needed according to the policy data at the next time of program deployment, any change made to an assigned program (e.g., removal of the program by a user) will be automatically corrected, e.g., an assigned program that was removed will be automatically made available for execution at the next time of deployment. Assigned programs are thus resilient.

Another significant concept that the prior art of record fails to teach or suggest is the concept of advertising, in which assigned programs may be made available for execution, before they are actually installed on the machine. In other words, when a program is advertised as available for execution prior to its installation, there is some data on the machine that makes the program in fact executable, even though at least some of the code that is needed to execute the program *is not yet actually installed on that machine*. Advertising, as defined in the specification, is a concept that is much more than simply letting a user know that an application can be *installed*, but rather refers to pre-configuring a local computer system with the information the system needs to *execute* program code. With advertised programs, any needed installation occurs automatically, as part of the execution process. Once installed, an assigned program is present on a system, and thus need not be advertised again unless and until that program is altered in some way (e.g., uninstalled or overwritten), whereby at the time of next deployment the assigned program will again be advertised as available for execution.

In general, advertising is generally accomplished by receiving and locally maintaining information about the program, (e.g., registry settings, icon data, file or object associations and so forth) prior to installation of the code required to run the program. For example, a shortcut that looks like a shortcut to any other executable program may be made to appear on the desktop and/or Start Menu of a computer system, even though the code that executes the program has not yet been installed on the computer system, and may will not be installed until actually needed to execute. Similarly, registry entries may be written to a system so that when execution is needed, (e.g., a file with a registered extension is double-clicked for opening), a corresponding program identity may be determined via the registry, whereby the appropriate program code may be automatically installed and then automatically executed. In other words, when execution is requested for an advertised program that is not already installed, the program (any needed code) is installed on demand, and thereafter automatically executed. As described below, advertising thus provides substantial benefits over the prior art of record, because among other things, programs are made available for execution, but not installed until execution is actually requested, thus avoiding unnecessary installations (which are expensive) for policy recipients that never request execution.

A third difference from the prior art of record is with respect to published programs. Published programs are those that the associations / policy data indicates as optional (e.g., specifically allowable but not mandatory) for a user and/or machine to have. Unlike assigned programs (including assigned programs that are advertised), published programs essentially have no presence on a machine (no registry entries, shortcuts, etc.) until activated in some manner, such as by requesting that a file be opened when that file has a file extension that corresponds to a published program, or requesting a published object. Published programs are not resilient like assigned programs, e.g., they may be removed or otherwise changed without being automatically restored.

Note that the above explanation is for informational purposes only, and should not be used to limit the claims, which are discussed below.

The §102 Rejections Based on Lang Are Improper as a Matter of Law

Before discussing the teachings of the art, applicants would like to comment upon the insufficiency of the Lang article as a reference. It is well established that a printed publication must be enabling. *Paperless Accounting, Inc. v. Bay Area Rapid Transit System*, 804 F.2d 659, 231 U.S.P.Q. 649 (Fed. Cir. 1986), *cert denied*, 480 U.S. 933 (1987). The Lang article, at best, provides brief, superficial summaries of products and functions, and thus is not close to being enabling as a reference that could legally support an anticipation or obviousness rejection of the present invention.

Notwithstanding, to the extent information can be gleaned from Lang, Lang clearly appears to be directed to numerous concepts that are fundamentally different from the described and claimed subject matter of the present invention. For example, Lang describes a point-to-point server model; with respect to the concepts of a client policies and associations, Lang is referring to a policy for centrally managing storage media and in particular, the *backing up and* *archiving* of client storage media. Lang, page 1, line 30 to page 2, line 5. Lang provides no clear teaching or suggestion of policy-based deployment of programs, and certainly does not describe a model in which policy recipients (e.g., network client users or machines) have programs assigned and published thereto, and/or advertised as available for execution.

In fact, in clear contrast to Lang, claim 1 is generally directed to associating two distinct types of programs with a policy recipient, namely those that are assigned as mandatory, and those that are published as optional. Lang, (which provides few if any details), at best mentions general software distribution, with no teaching or suggestion of mandatory and optional programs, let alone any distinction between such deployed programs, and thus fail to disclose the concept of assigned and published programs associated with policy recipients. New claims 27-43 similarly recite such concepts.

By law, in order to support an anticipation rejection, the Office action is required to show that each and every element of the claimed invention is disclosed in a single reference, and that each element is arranged as in the claim. Lang, which simply does not disclose or in any suggest the concept of assigned and published programs, clearly fails these requirements. As a result, the rejections are improper as a matter of law, and the claims (including those newly added) are patentable over Lang. Reconsideration and withdrawal of the rejections of pending claims 1-7, and 10-14 based on Lang is respectfully requested.

Applicants also submit that the dependent claims are separately patentable over the prior art of record for numerous other significant reasons. However, because Lang is so seriously deficient in its teachings with respect to the independent claims of the present invention, the rejections of the dependent claims will not be separately addressed at this time for purposes of brevity.

The §103(a) Rejections Based on Lang and Stupek Are Improper as a Matter of Law

By law, in order to establish *prima facie* obviousness of a claimed invention, all of the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). In addition, "all words in a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970). Further, if prior art, in any material respect teaches away from the claimed invention, the art cannot be used to support an obviousness rejection. *In re Geisler*, 116 F.3d 1465, 1471, 43 USPQ2d 1362, 1366 (Fed Cir. 1997).

Applicants submit that not only do Lang and Stupek fail to disclose the recited limitation of advertising in claims 15 and 24, (e.g., advertising a program as available for execution prior to its installation), but moreover submit that both Lang and Stupek directly teach away from such a concept. For example, to the extent anything can be discerned from Lang, the only software distribution referred to in Lang appears to be simple remote software installation, in which personal computers are woken up if off-hours, and any software is fully installed at that time. Lang, page 1, lines 21 to 25. This means that software installation in Lang is unrelated to software execution, e.g., Lang does not perform on-demand installation in response to an *execution* request, but rather, as is typical, performs deferred installation in response to an *installation* request. Lang thus teaches the desirability of complete installation during off-hours that is entirely unrelated to an on-demand need to execute a program, which directly teaches away from advertising, in which programs need not be installed until actually needed for execution, e.g., on demand in response to an execution request.

Similarly, like Lang, no reasonable interpretation of Stupek teaches or even suggests the concept of advertising a program as available for execution prior to its installation. Instead, like

Lang (to the extent Lang discloses anything), Stupek handles installation separately from execution, given the plain and ordinary meaning of the terms "execute" and "install" (and their variants) with respect to programs, which is also consistent with the usage of these terms in the specification. Stupek may show that an update can be installed, and then sometime later installs that update following an *installation* request, but Stupek simply does not do the installation in response to a request to execute the program.

In fact, Stupek teaches directly away from the concepts of assigned programs and/or advertising programs as available for execution. More particularly, Stupek teaches that an update is installed only when a user requests to download it, which is the opposite of assigned programs that a user policy recipient has no choice but to have. Also Stupek teaches that the requested update is only installed when the server later gets around to transmitting it, at midnight, (Stupek column 10, lines 16-18), which is the opposite of advertising a program to facilitate on-demand installation in response to a request to execute a program. Stupek's deferred installation is only about installation, and has nothing to do with any request to *execute* a program, nor could it reasonably be interpreted that way. By way of example, requesting that a program be executed such as during typical daily working hours and then having to wait until midnight (or even some other deferred time) before it was even installed would be nonsensical.

In sum, a reasonable reading of Stupek in its entirety makes it clear that Stupek's installation is not in response to a request to execute a program at all, but rather only in response to a request to update (download and install) a program. Although the Office action considers Stupek's update notification to be "advertising" in a general sense, advertising that an application is available for installation is not the same as (or even analogous to) advertising a program as available for execution. While the terms "notification" and "advertising" are unfortunately similar in certain contexts, the meaning of notification in Stupek is wholly different from the meaning of advertising as used in the claims and defined in the specification. When read in their full contexts, Stupek, which only notifies a user that an update is available for installing (and nothing more), and the claimed invention, which advertises a program as available for execution prior to installing it, are extremely different concepts.

Thus, even if somehow permissible to combine Stupek with Lang (which applicants submit it is not, as discussed below), the claim limitations are not found by such a combination. Further, Lang and Stupek both teach away from the claimed invention, and thus cannot be used to support an obviousness rejection as a matter of law. For at least these reasons, applicants submit that the rejections of claims 8, 9 and 15-24 are improper as a matter of law, and respectfully request reconsideration and withdrawal of the rejections.

Also as a matter of law, obviousness may not be established using hindsight obtained in view of the teachings or suggestions of the applicants. *W.L. Gore & Assocs., Inc. v. Garlock, Inc.*, 721 F.2d 1540, 1551, 1553, 220 USPQ 303, 311, 312-13 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984). To guard against the use of such impermissible hindsight, obviousness needs to be determined by ascertaining whether the applicable prior art contains any suggestion or motivation for making the modifications in the design of the prior art article in order to produce the claimed design. The mere possibility that a prior art teaching *could* be modified or combined such that its use would lead to the particular limitations recited in a claim does not make the recited limitation obvious, unless the prior art suggests the desirability of such a modification. *See In re Gordon*, 733 F.2d 900, 902, 221 USPQ 1125, 1127 (Fed. Cir. 1984).

As discussed below, the prior art contains nothing at all that supports the conclusion that any of the claims are obvious. Instead, the Office action has relied on the lack of detail in the prior art's teachings to speculate that the prior art somehow could be combined or modified to reach the present invention, as discussed below, which is unquestionably contrary to law. In the present case, it is clearly evident that the Examiner can only have used impermissible hindsight gleaned from applicants' own teachings to locate Lang and Stupek, and thereafter attempted to fit their teachings into applicants' invention, even though neither teach assigned, published and/or advertised programs, and both teach away from the present invention as claimed.

In fact, the alleged motivation in the Office action for combining Lang with Stupek itself leads away from assigned programs, and also confuses the concept of performing an installation with executing an already installed program. More particularly, the Office action refers to an "opt to launch feature which enables a client with the option not to upgrade or to upgrade and do so immediately." However, even if this contention was true, (which applicants submit is not found in Lang and/or Stupek), then this would teach directly *away* from assigned programs (including upgrades) which are mandatory, not optional. In addition, as discussed above, there is no immediate launch of anything taught in Lang or Stupek, as in both references, installation is deferred at the server, (in off-hours in Lang, and at midnight in Stupek), and thus such installation is neither in response to execute a program, nor on demand.

For at least these additional significant reasons, applicants submit that the §103(a) rejections are improper as a matter of law. Reconsideration and withdrawal of the rejections of claims 8, 9 and 15-24 is respectfully requested.

CONCLUSION

In view of the foregoing remarks, it is respectfully submitted that claims 1-22 and 24-43 of the present application should not be rejected on the art or otherwise, and that the application is good and proper form for allowance. A favorable action on the part of the Examiner is earnestly solicited.

If in the opinion of the Examiner a telephone conference would expedite the prosecution of the subject application, the Examiner is invited to call the undersigned attorney at (425) 836-3030.

Respectfully submitted,

Albert S. Michalik

Albert S. Michalik, Registration No. 37,395
Attorney for Applicants
Law Offices of Albert S. Michalik, PLLC
704 - 228th Avenue NE
Suite 193
Sammamish, WA 98074
(425) 836-3030 (telephone)
(425) 836-8957 (facsimile)

APPENDIX A

(Version with markings to show changes made in the related applications section beginning at page 1, line 4)

RELATED APPLICATIONS

This application is related to the following United States Patent applications, all of which are filed on the same day and assigned to the same assignee as the present application:

["Method and System for Advertising Applications" serial no. ___________, hereby incorporated by reference herein in its entirety;

"Class Store Schema" serial no. ___________;

"Method and System for On-Demand Installation of Software Implementations" serial no. ___________; and

"Software Implementation Installer Mechanism" serial no. ___________]

"Method and System for Advertising Applications" serial no. 09/158,967, now U.S. Patent No. 6,345,386, hereby incorporated by reference herein in its entirety;

"Class Store Schema" serial no. 09/158,023, now U.S. Patent No. 6,389,589;

"Method and System for On-Demand Installation of Software Implementations" serial no. 09/158,022 and

"Software Implementation Installer Mechanism" serial no. 09/158,021, now U.S. Patent No. 6,418,554.

Appendix B

(marked up copy of the claims amended herein)

1. (Amended) In a computer network, a method of [automatically] deploying [an application] a plurality of programs to a policy recipient, comprising [the steps of], maintaining

an association between [the application] each program and the policy recipient, at least one association corresponding to a program that is assigned as mandatory for the policy recipient to have on a computer system associated with that policy recipient, and at least one other association corresponding to a program that is published as optional for the policy recipient to have, determining a time [to apply information of the application] for deploying each assigned program to the policy recipient[,] and making [the application] that program available to the policy recipient at that time, and deploying each published program to the policy recipient when the published program is activated via the computer system.

2. (Amended) The method of claim 1 wherein the policy recipient is a user, and wherein [the step of] determining a time [to apply the application] for deploying each assigned program to the policy recipient includes [the step of] detecting that the user is logging on to the network.

3. (Amended) The method of claim 1 wherein the policy recipient is a machine comprising the computer system, and wherein [the step of] determining a time [to apply the application] for deploying each assigned program to the policy recipient includes [the step of] detecting that the machine is connecting to the network.

4. (Amended) The method of claim 1 wherein [the step of] determining a time [to apply the application] for deploying each assigned program to the policy recipient includes [the step of] polling machines in the network.

5. (Amended) The method of claim 1 further comprising, [the steps of] detecting a request to run [the application] a selected one of the assigned programs, accessing the registry to locate the [application] selected assigned program, and executing the [application] selected assigned program.

6. (Amended) The method of claim 1 wherein [the step of] maintaining an association between [the application] each assigned program and the policy recipient comprises [the step of] storing [a] at least one script in association with policy information for that policy recipient.

7. (Amended) The method of claim 6 wherein [the step of] storing [a] each script comprises [the step of] maintaining a centralized directory of network objects.

8. (Amended) The method of claim 1 wherein [the step of] making [the application] the program available to the policy recipient comprises [the step of] advertising the [application] program as available for execution prior to installation of program code used to execute the program on the computer system, and further comprising, in response to a request to execute the program, installing the program code at the computer system and executing the program code.

9. (Amended) The method of claim [1] 8 wherein [the step of] advertising the [application] program as available includes [the step of] adding an application shortcut to a user

profile such that a user perceives the program as executable prior to installation of program code used to execute the program on the computer system.

10. (Amended) The method of claim 1 wherein [the step of] making the [application] program available to the policy recipient comprises [the step of] writing information to a registry associated with the policy recipient such that a request related to executing the program prior to installation of program code used to execute the program on the computer system will have corresponding information in the registry that identifies the program for automatic installation and execution thereof.

11. (Amended) The method of claim 10 wherein [the step] writing information to [a] the registry comprises [the step of] maintaining in the registry an association between the [application] program and class identifier information.

12. (Amended) The method of claim 10 wherein [the step] writing information to [a] the registry comprises [the step of] maintaining in the registry an association between the [application] program and a file extension.

13. (Amended) The method of claim 12 further comprising [the steps of], detecting a user request to open a file, determining the file extension and accessing the registry to locate [the application] a corresponding program associated with the file extension, and opening the file with the [application] corresponding program.

14. (Amended) The method of claim 13 further comprising [the steps of], determining if the [application] program is locally installed, and if not installed, installing the [application] program and executing the program to open the file.

15. (Amended) In a computer network, a method of automatically deploying an [application] assigned program to a policy recipient, comprising [the steps of], generating a script having information therein corresponding to the [application] assigned program, associating the script with a policy, [and] applying the policy to the policy recipient by processing the script to advertise the [application] assigned program as available [thereto] for execution by the policy recipient prior to installation of program code needed to execute the program, such that the program is installable on demand in response to a request to execute the program.

16. (Amended) The method of claim 15 wherein the policy recipient is a user, and wherein [the step of] applying the policy to the policy recipient is performed as part of a user network logon process.

17. (Amended) The method of claim 15 wherein the policy recipient is a machine, and wherein [the step of] applying the policy to the policy recipient is performed as part of a re-boot process of the machine.

18. (Amended) The method of claim 15 wherein [the step of] associating the script with a policy comprises [the step of] maintaining the association in a centralized directory of network objects.

19. (Amended) The method of claim 15 wherein [the step of] applying the policy to the policy recipient includes [the step of] adding an application shortcut to a user profile to advertise the program as available for execution.

20. (Amended) The method of claim 15 wherein [the step of] applying the policy to the policy recipient includes [the step of] writing information to a registry associated with the policy recipient to advertise the program as available for execution.

21. (Amended) The method of claim 20 wherein [the step] writing information to a registry comprises [the step of] maintaining in the registry an association between the [application] program and class identifier information.

22. (Amended) The method of claim 20 wherein [the step] writing information to a registry comprises [the step of] maintaining in the registry an association between the [application] program and a file extension.

24. (Amended) In a computer network, a system for deploying [applications] programs to policy recipients, comprising, a centralized store of policy objects, at least one of the policy objects having an advertise script associated therewith, the advertise script including

[application] program information of at least one [application] program for assigning to policy recipients, and a mechanism for applying policy objects to a policy recipient, wherein at least one of the policy objects applied to that recipient has the advertise script associated therewith, the mechanism processing the script to advertise at least one [application] assigned program to the policy recipient such that the assigned program is advertised as available for execution by the policy recipient prior to installation of program code needed to execute the program.